УДК 004.8

МОЛЧАНОВСЬКИЙ О.І.,
КОВТУН Д.І.

СЕМАНТИЧНИЙ ПОШУК ТЕКСТІВ НОВИН

У статті розглянуто представлення семантичної інформації текстів на природних мовах у вигляді множини триплетів: «суб'єкт-предикат-об'єкт». Представлено метод отримання множини триплетів з тексту та побудову деревовидної ієрархії триплетів. Запропоновано метод пошуку текстів по сформованій множині триплетів.

In this paper we examine representation of semantic information of natural language text in a view of a set of triplets: "subject-predicate-object". We present the method of how to get the triplet set from a news text and how to build a tree hierarchy of obtained triplets. The method of searching triplet within created set is proposed.

Вступ

Зважаючи на значне зростання обсягів текстової інформації та складної структурованості природно-мовних текстів, семантичний аналіз текстів являє собою актуальну проблему, особливо в останні 15-20 років, коли намітилася тенденція до інформатизації суспільства.

Далеко не кожен користувач інформаційних засобів здатний чітко висловити свої інформаційні потреби. Для цього необхідний певний рівень знань в конкретній області, певну інформацію з якої і бажає отримати користувач.

Ця проблема стала приводом до появи систем семантичного аналізу текстів. Семантичний аналіз тексту – це процес отримання структурованої інформації з тексту на природній мові. Фактично такий аналіз здійснює кожна людина, навіть не замислюючись над цим.

Об'єктом нашого дослідження є побудова системи пошуку текстів новин на природній мові з використанням семантичної інформації самих текстів та запиту користувача системи.

Існуючі пошукові системи

Розглянемо існуючі пошукові системи, які спеціалізуються на темі, яка нас цікавить – пошук новин. Їх досить багато, тому ми виділили три найбільш популярні.

Пошукова система WebCrawler (webcrawler.com) – це метапошуковий двигун, який поєднує в собі кращі результати пошуку від Google, Yahoo!, Bing Search, Ask.com, About.com, MIVA тощо. Він надає користувачам можливість шукати зображення, аудіо, відео, новини, жовті та білі сторінки.

Пошукова система Excite (excite.com) вигідно відрізняється від інших пошукових вузлів тим, що дозволяє вести пошук англійською мовою в службах новин і публікує огляди вебсторінок. База даних цього сайту складається з більш ніж 50 млн. сторінок з індексацією за повним текстом.

Система RedTram (redtram.com) – пошукова система новин, що дозволяє користувачам швидко знаходити найсвіжішу інформацію на будь-яку тему, що їх цікавить. Унікальність цієї пошукової системи полягає у багатомірності критеріїв, що застосовуються одночасно для пошуку та відображення новин: тематика, регіон, мова, дата. Синхронізація вказаних критеріїв дозволяє користувачу максимально конкретизувати запит на пошук та отримати в результаті саме ті новини, які його цікавлять.

Проведемо простий експеримент з системою RedTram. На головній сторінці сайту відображається пошуковий рядок та декілька останніх новин. В нашому прикладі це була новина із заголовком «Eggs: the new super-food?» (рис. 1).

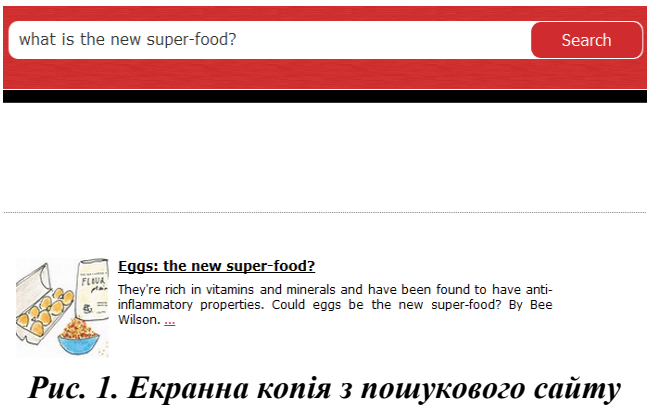

Рис. 1. Екранна копія з пошукового сайту RedTram

Ми поставили системі питання "What is the new super-food?" (ввели його у пошуковий рядок), на яке людина легко могла б отримати відповідь, маючі інформацію про дану новину. Але результати пошуку виявились невтішними (рис. 2).

Рис. 2. Результати запиту

Тож постає просте питання: чому ж ці популярні пошукові системи так погано справляються зі, здавалось би, простим завданням?

Розробки в області семантичного аналізу тексту пов'язані з областю штучного інтелекту, що робить акцент на смисловому розумінні тексту [1]. В даний час успіхи в цьому напрямку досить обмежені. Розроблені семантичні аналізатори володіють високою обчислювальною складністю і неоднозначністю видаваних результатів [2]. Зараз інтенсивно розвивається напрямок, пов'язаний із застосуванням різних видів онтологій для цілей повнотекстового пошуку в електронних колекціях документів [3].

Семантична інформація на основі триплетів

Однією з найпоширеніших мов представлення інформації за допомогою онтології [4] є Resource Description Framework (RDF). Базовою структурною одиницею RDF є трійка (або триплет), який складається з суб'єкта, предиката і об'єкта (рис. 3).

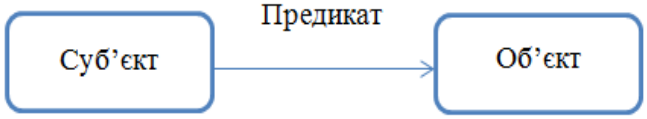

Рис. 3. RDF-триплет

Одне з рішень перетворення тексту на природній мові у RDF-представлення наведено в роботі [5]. Пропонується будувати семантичну модель та дерево розбору за допомогою програми-лексера та програми-парсера, яка розуміє граматику вибраної мови. Лексер – програмний модуль, що здійснює розбір коду за заданими граматичними правилами та генерує послідовність нетермінальних символів мови. Парсер – програмний модуль, який, базуючись на інформації, що надходить з лексеру, по заданим правилам будує абстрактне дерево розбору. Подальший розбір дерева полягає у виділенні необхідних та важливих для поставленої задачі вузлів та листків синтаксичного дерева. Робота парсера полягає у трансформації інформації, яка надходить, у виді триплетів, якщо ця інформація має цінність для поставленої задачі.

Триплети не єдиний інструмент аналізу тексту – вони завжди використовується з іншими методами. Наприклад, при структуризації даних у гіпертекстових масивах, триплети використовуються як допоміжний механізм прихованого семантичного аналізу [6]. Це метод обробки природної мови, що дозволяє проаналізувати взаємозв'язок між колекцією документів і термінами, які в них зустрічаються.

Побудова триплетів

Повертаючись до нашої мети, нагадаємо, що ми хочемо максимально ефективно використати модель представлення «суб’єкт – предикат – об’єкт» для отримання якісного аналізу вихідного тексту новин. Для цього ми використовуємо openNLP (opennlp.apache.org) парсер, який надає змогу визначати частини речення (підмет, присудок тощо). Таким чином, кожне речення – це об’єкт, який представляє собою деревовидну структуру, де кожна з гілок дерева представляє фразу, кожний листок – слово.

Для того, щоб визначити частини речення в побудованому дереві використовуються правила розбору речення. Правила поділяються на дві групи – правила рівня фраз та правила рівня слова. Всі правила мають однакову структуру і складаються з двох частин – тегу та типу. Результатом використання правила до фрази чи слова буде частина речення (підмет, присудок, додаток), якщо слово чи фраза такою є. Тег – це мітка, що проставляється в дереві речення, яке повертає парсер бібліотеки openNLP; він визначає, що представляє собою фраза або слово. Наприклад, фраза може мати наступні теги :

ADVP – Adverb Phrase;
NP – Noun Phrase;
VP – Vereb Phrase;
NN – Noun, singular or mass.

Тип також проставляється парсером на гілках дерева. Він представляє собою скорочену інформацію про гілку у вигляді скорочень. Тип відображає чи має слово або фраза відноситися до структури, яка може вміщувати в собі частини речення. Тип може бути відсутній для фрази чи слова.

Для знаходження частин речення, кожна гілка речення після парсеру, перевіряється правилом. Якщо фраза задовольняє правилам запускається алгоритм розбору фрази. Інакше, вона

пропускається, як не значима для речення. Результатом розбору речення – є набір триплетів.

Фраза являє собою частину речення, тому також має деревовидну структуру. Дане дерево обходиться пошуком вглиб, коли не значимі слова опускаються, значимі фіксуються на кожному з рівнів дерева. Після чого йде формування частини речення, яка буде фігурувати у триплетах. Це виконується за допомогою зворотного обходу дерева – від листків до кореня. При чому значимі слова кожного рівня утворюють набір фраз, які найкращим чином відображають суть фрази. Результатом розбору фрази буде одна частина триплету.

Після розбору всіх фраз речення, запускається формування триплетів. Речення являє собою набір слів з позначками, які відносять кожне слово до якоїсь частини речення: якщо слово відноситься до підмету, воно буде поставлено на перше місце триплету (суб'єкт), якщо присудок – на друге (предикат), якщо додаток – на третє місце (об'єкт). Якщо в реченні присутнє декілька додатків, то триплет даного речення дублюється із новим додатком в якості об'єкту триплету.

Однією з проблем формування триплетів по реченнях є обробка займенників. Необхідно вміти замінювати займенники, які виконують роль підмету, на відповідні іменники. Ми використовуємо простий наступний метод. Для кожного підмета речення в триплеті додатково зберігається його тип (іменник чи займенник). Коли всі триплети тексту сформовані, відбувається послідовна заміна займенників на іменники, які зустрічались безпосередньо вище в тексті.

На основі сформованих триплетів формується дерево, яке дає можливість відслідковувати, як змінюється тема тексту, коли розповідь від однієї сутності переходить до іншої (від підмета до додатку, наприклад). Дерево будується наступним чином. Для даного списку триплетів, задається глибина, на якій може бути зв'язок між додатком одного речення та підметом другого. Величина глибини визначається емпіричним шляхом для тематики текстів(для новин даний показник складає 3). Потім за один прохід по списку будується дерево триплетів. Кожен триплет може мати дочірні вузли, при умові що фокус розповіді переміщується з додатку поточного на підмет наступного триплету. Тобто якщо в одному триплеті слово було додатком, а в іншому це ж слово виступає в ролі підмету, то буде створено зв'язок: перший триплет буде виступати в ролі батьківського, другий – в вигляді дочірнього вузла.

Розглянемо приклад розбору речення на англійській мові: "The Istanbul meeting faces a mass of dilemmas contradictions and complexities with no clear way forward". Парсер визначив наступні частини речення (табл. 1).

Табл. 1. Частини речення для фраз речення-прикладу

Фраза	Частина речення
Istanbul meeting	SUBJECT
faces	PREDICATE
mass of dilemmas	OBJECT
mass of contradictions	OBJECT
mass of complexities	OBJECT

Множина сформованих триплетів виглядає так:

- subject='Istanbul meeting', subjectType='NP', predicate='faces', object='mass of dilemmas';
- subject='Istanbul meeting', subjectType='NP', predicate='faces', object='mass of contradictions';
- subject='Istanbul meeting', subjectType='NP', predicate='faces', object='mass of complexities'.

Речення складають тексти і у тексті завжди є головна думка та другорядна. Переведення всього текст у триплети не є ефективним. Нам необхідна ієрархія триплетів, яка виявить які речення необхідно зберегти у триплети. Речення на вершині ієрархії будуть збережені для майбутнього використання, речення ж третього та наступних рівнів будуть проігноровані.

Розглянемо невеликий приклад: "I am playing in a game and talking with girl. Game is called Duty. Duty is a game. Duty is a shooter. Girl is playing with baby. Baby is very small and lies into bad. Mom went to the sea. Sea is warm".

Ієрархія побудованих триплетів представлена на рисунку 4:

```
subject='I', predicate='am playing', object='game'
    subject='Game', predicate='is called', object='Duty'
subject='I', predicate='am talking', object='girl'
    subject='Girl', predicate='is playing', object='baby'
        subject='Baby', predicate='is', object='small'
        subject='Baby', predicate='lies', object='bad'
subject='Duty', predicate='is', object='game'
subject='Duty', predicate='is', object='shooter'
subject='Mom', predicate='went', object='sea'
    subject='Sea', predicate='is', object='null'
```

Рис. 4. Ієрархія триплетів

Частота входжень кожного підмета представлена у таблиці 2.

Табл. 2. Частота підметів у тексті

Фраза	Частина речення
Duty	0.335
I	0.225
Girl	0.11
Baby	0.11
Mom	0.11
Sea	0.11

Пошук триплетів

Отримані триплети в першу чергу використовуються для пошуку інформації. В якості критерію оцінки важливості інформації застосовується частота входжень підметів речення у триплетах та рівень ієрархії. Сам пошук реалізується наступним чином.

Із запиту, який наданий користувачем, формується спеціальний триплет, у якого не вистачає однієї частини з «суб'єкт – предикат – об'єкт» представлення. Мета пошуку – по двом ключовим частинам знайти третю, таким чином однозначно визначити до якого тексту відноситься триплет та вивести зміст новини користувачеві.

Для прикладу візьмемо вже знайоме нам речення з першого запиту "What is the new super-food?". Парсер визначив наступні частини речення (табл. 3).

Триплет-запит має вигляд:

- subject=null, predicate='is', object='super-food'.

Табл. 3. Частини речення для фраз запиту-прикладу

Фраза	Частина речення
is	PREDICATE
super-food	OBJECT

В цьому випадку ми будемо шукати підмет за присудком та додатком.

Висновки

В даній роботі був запропонований метод пошуку текстів новин, в основі якого лежить представлення текстів природної мови у вигляді семантичних триплетів, використовуючи формат RDF. Майбутня робота буде спрямована в бік деталізації запропоновано підходу та реалізації конкретної системи пошуку текстів новин в мережі Інтернет.

Перелік посилань

1. Apte, C., Damerau, F.J., Weiss, S.M., Automated learning of decision rules for text categorization. ACM Transactions on Information Systems 12, 3, 233–251., 1994
2. Dagan, I., Karov, Y., Roth, D., Mistake-driven learning in text categorization. In Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing (Providence, US, 1997), pp. 55–63., 1997
3. Дерецкий В.А. Підхід до автоматичної побудови тематичної онтології документу для удосконалення інформаційного пошуку – 2005. – № 3. – С. 76–82
4. T. R. Gruber. What is an Ontology? [Электронный ресурс] / T. R. Gruber. – Режим доступа: http://www-ksl.stanford.edu/kst/what-is-an-ontology.html
6. Зараковский А.В., Клименков С.В., Ткаченко Н.И., Харитонова А.Е. Основные принципы рещения задачи преобразования объектно-ориентированного кода в формат RDF средствами семантического анализа. // Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики, 2011, № 2 (72).
7. А.В.Заболеева-Зотова, Н.А.Козлова, А.Ю.Пастухов, П.В.Сердюков, С.А.Чернов. LSA в гипертекстовых масивах. [Електронний ресурс] // Режим доступу: http://transfer.eltech.ru/innov/archive.nsf/0d592545e5d69ff3c32568fe00319ec1/88dcc833ef0efed8c3256a76004c6bbd?OpenDocument